ATTORNEY DOCKET NO.: CIS01-23(4696)

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

| | |
|---|---|
| Applicant: | Suchitra Raman |
| Serial No.: | 10/044,212 |
| For: | METHODS AND APPARATUS FOR POOLING AND DEPOOLING THE TRANSMISSION OF STREAM DATA |
| Filing Date: | November 20, 2001 |
| Examiner: | Gillis, Brian J. |
| Art Unit: | 2141 |
| Conf. No.: | 7901 |



Certificate of Mailing Under 37 C.F.R. §1.8

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: **MAIL STOP AMENDMENT,** Commissioner for Patents, PO Box 1450, Alexandria, Virginia 22313-1450 on:

Date: May 10, 2005

By: Mary A. Maietta
(Typed or printed name of person mailing Document, whose signature appears below)

Signature: Mary A. Maietta

**MAIL STOP AMENDMENT**
Commissioner for Patents
PO Box 1450
Alexandria, Virginia 22313-1450

## AFFIDAVIT UNDER 37 CFR 1.131

1. I, Suchitra Raman, am an inventor of "Methods and Apparatus for Pooling and Depooling the Transmission of Stream Data" now before the U. S. Patent Office as Application Number 10/044,212.

2. It is my belief that inventorship in the above-identified patent application is correct.

3. Prior to August 27, 2001, I conceived of the invention of "Methods and Apparatus for Pooling and Depooling the Transmission of Stream Data" now before the U. S. Patent Office as Application Number 10/044,212.

4. Prior to August 27, 2001, I reduced to practice the invention described and claimed in Application Number 10/044,212, now pending in the U.S. Patent Office, as evidenced by the attached document entitled "Patent and Details (#131488), Packet Pooling for Efficient Stream Distribution over Packet-switched Networks".

5. All of the statements made herein are of my own knowledge and are true, these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment or both, under § 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application and any patent issuing thereon, or any patent to which this verified statement is applied.

5/6/05
Date of Signature

By: 
Suchitra Raman

WORKING COPY



# #131488: Packet Pooling for Efficient Stream Distribution over Packet-switched Networks

## Status Info

**Entered:** 08-May-2001
**Modified:** 08-May-2001
**State:** Approved. In Process.

**Serial #:** ---
**File Date:** ---

**U.S. Pat #:** ---
**Issue Date:** ---
**View patent:** *Not available.*

**Current Inventors:**
Suchitra Raman (suraman).

**History:**
*No additional history currently available.*

## Full Details

**Inventors:** (As submitted) Suchitra Raman (suraman) **Work Phone:** 781 663-8304 **Manager:** elifuchs **Department:** CDN Engineering **Division:** SP PCIPS **Site:** WALTHAM **Info:** View in Cisco Directory

**Background:** Content distribution networks (CDNs) use application-layer multicast to efficiently distribute live and stored content to edge nodes. Live media can be distributed using RTP or other equivalent protocol (e.g., RealNetworks' RDT protocol, Microsoft's MMSU/MMST protocols). This invention describes how to efficiently packetize RTP stream data in a method suitable for efficient transport through such an application-layer multicast network.

Live stream distribution calls for an efficient one-to-many communication method. One-to-many distribution at layer 3 of the traditional protocol stack can be accomplished using IP multicast

routing. However, today's IP routing infrastructure does not ubiquitously support this extension to the IP service model. This has given rise to the "overlay" model in which intermediate appliances placed at selective network locations perform packet replication to provide efficient one-to-many transport through the network. (Note: Packet replication is also referred to as stream splitting when used in stream distribution.)

An application-layer multicast network consists of appliance nodes responsible for splitting input streams to multiple downstream appliances, and so on, until the stream ultimately reaches all its clients that requested it. The application-layer multicast topology forms an overlay routing layer on top of the existing unicast IP routing layer. Here, we describe packetization technique at the appliance nodes that leads to efficient stream transport through the overlay network.

A stream splitting appliance has many input streams each at a certain known average (encoded) bitrate. Many CDN workloads consist of large numbers of "thin" (low bitrate) streams such as 64 kbps audio sourced by online radio stations, audiocast, narrowband video (<300 Kbps). The exact relationship between encoding bitrate and average packet size depends on the specific encoding algorithm. However, in general, low fidelity streams consist of small datagrams (100-300 byte packets).

This common workload that comprises many small packets has two problems:

(i) Forwarding many low bitrate (thin) streams each with small datagrams puts a heavy load on each forwarding appliance because of interrupt handling and header processing on a per-packet basis.

(ii) Such a workload also has consequences on the lower layer forwarding devices such a routers and L2 switches in the network path. Most L2 and L3 devices do not handle a large number of small datagrams optimally because of increaased packet processing and route lookup overhead.

**Summary:** The main idea in stream pooling is to aggregate or pool multiple RTP packets, from independent streams, that are destined along the same path into the same IP datagram. We call such an aggregate datagram a "packet pool". Both the interrupt handling overhead and header processing can be optimized by this aggregation method. This method requires the following issues to be addressed:

(1) What criteria to use to pool packets from potentially independent flows into the same packet pool?

(2) How should an intermediate node that is presented with a packet pool forward the pool datagram?

(3) How should a node separate or reconstruct individual packets from a packet pool? We call this "de-pooling."

---

We outline how our stream pooling technique handles these issues.

(1) Packet Pool Construction:

How does an appliance detect packets that would benefit from being transported in a single IP datagram?

(a) Which packets should be pooled?
A simple implementation would apply stream pooling arbitrarily to stream packets of any flow as long as they are known to travel together to the next stream forwarding node. In other words, the stream aggregation step uses a lookahead of one to determine how much sharing of stream paths occurs between the diferent flows.

If more information about the overlay path taken by the different flows is known, a more elaborate aggregation scheme is possible. The generalized lookahead algorithm is described below. This requires looking up the entire forward path of a given stream by inspecting the routing tables. The details of this lookup depend on the exact routing protocol being used. This lookup is simplest when the overlay nodes use a link-state multicast routing protocol, such as M-OSPF, based on stream IDs. In this case, the entire path of appliance nodes taken by a stream is available from the OSPF link state database available at each stream appliance. We assume that the entire appliance network is using a single OSPF area. Aggregation is based on the amount of correlation between descendant sets.

(b) What are the timing constraints of the pooling process?
More packets an appliance can aggregate, the greater the performance advantages. However, a wait time may be required to gather enough packets, Hence, what is the maximum per-hop delay that an appliance impose on a packet while making pooling decisions. Multiple criteria are possible here.

(a) Target candidate size: In the first scheme, each stream forwarding appliance reads a constant number of packets, constructs a stream aggregates and forwards them along. Pooling is performed only with

those stream packets that are much smaller than an MTU (maximum transmission unit). This works well when the appliance has a large number of thin streams in its input queue. Except for the additional processing on the aggregator (and de-aggregator) node, this method does not impose any additional delays on stream packets.

(b) Max. delay: This variant is better suited when the bandwidth is not high enough and the aggregation method can incur an additional delay.

```
Wait time = MIN(T_p, T_d);        T_p = Time to get p packets
                                  T_d = Max. delay.
```

(2) Forwarding a packet pool:

A node that is replicating/forwarding packets to downstream nodes is not strictly required to inspect the packet pool prior to forwarding. However, if the overhead of unnecessary transmissions of extraneous RTP packets within the packet pool to downstream nodes is unacceptable, a node may de-pool the packet.

(3) De-pooling:

This is required only at the edges of the network. I.e., if an appliance is a leaf node in the overlay distribution network, it needs to inspect the packet pool and separate them into individial packets before delivering to a client.

However, if de-pooling is necessary within the distribution network, it is done based on the packet pool header in the extended RTP header. The packet pool header is an index that provides information on each of the contained packets.

==========

**Advantages:** Current stream splitting products do not take advantage of shared paths to aggregate packets. As a result, their per-appliance stream capacity is small. With stream pooling, each appliance can deliver greater stream throughput.

**Cisco use:** Cisco's CDN products can benefit from this technique. The recently proposed Stream Engine product can use stream pooling for efficient distribution. Small packets can otherwise adversely impact the stream processing performance of each Stream Engine.

**Method of detecting use by other companies:** There is no externally visible effect of this technique. However, any product that advertises high stream splitting or stream forwarding capability must be examined for infringement. RealServer splitters are in a position to use it for efficient stream distribution, but do not currently do so in their RealSystem iQ network.

**Previous public use:** None.

**First written record date:** 08-May-2001

**First written record URLs:** ---

**Supporting docs URLs:** ---

**Notes:** ---

## Review Information

**Progress:**

| Review Group | Reviewer Users | Status |
|---|---|---|
| SPLOB-PCIPS-CST | mcieslak, **otoole***, rnair | **Reviewed** |

*Designated Reviewer.

This idea has been **approved**. Please see the Status Info above for its current status.

**Comments:**

**User ID:** otoole **Date:** 5/31/2001 21:16

REASON FOR APPROVAL:
Now I understand this one and why it isn't the same as MPLS labels.

OTHER NON-CISCO USERS OF THIS INVENTION:
Inktomi/FastForward, ExtremeNetworks, Alteon

DETECTION OF NON-CISCO USE:
Probably documented, but even if not, would be obvious to sniffer.